Dump

Objective

In this projectlet, files are explored further - treating them as binary streams. In addition, a template for command line tools is developed.

Conventions

Let us adopt the following conventions for all command line utilities:

Id	Convention
1	If invoked with no command line argument, the utility shall print a help message
2	Utility shall report a version number in the form: name-major.minor
3	-h or —help shall be supported switches to ask to print the help message
4	-v or—verbose shall be supported switches to be verbose ie provide diagnostic details

User needs and requirements

This projectlet is to fulfill the need of software engineers.

Id	Need/Requirement
1	The utility show the contents of a file in the form of a dump as well as in text
2	In the text form, non printable data shall be shown as "."
3	In each line 1 block of data shall be shown.
4	The utility shall support a default block size of 32.
5	The utility shall recognize the switch -b —blocksize to override the block size. Valid block sizes are 8, 16 and 32
6	The default binary output shall be in hexadecimal.
7	The utility shall recognize the switch -o —octal to provide an octal output.
8	The utility shall display the file offset in each line.
9	Each file shall be preceded by a list of the full path of the file name, its size and creation date.

Example usage

The following is an example output where the dump is in hexadecimal. The first column ie the file offset is printed in hexadecimal form. The data is largely binary though there are printable ascii characters.

```
Blocklength is 16
File: /Users/rajasrinivasan/Projects/go/dump/dump
00000000 cffaedfe070000010300000002000000 Ïúíþ............
00000010 0a000000f80800000100000000000000 ....ø...........
00000020 19000000480000005f5f504147455a45 ....H...__PAGEZE
00000030 524f0000000000000000000000000000 RO..............
00000040 00000001000000000000000000000000 ................
00000050 00000000000000000000000000000000 ................
00000060 00000000000000001900000078020000 ............x...
00000070 5f5f5445585400000000000000000000 __TEXT..........
00000080 000000010000000000f0130000000000 .........ð......
00000090 000000000000000000f0130000000000 .........ð......
000000a0 07000000050000000700000000000000 ................
000000b0 5f5f7465787400000000000000000000 __text..........
000000c0 5f5f5445585400000000000000000000 __TEXT..........
```